



| | | |
|--|-----------------------------------|--|
|   | ATL Transformation Example | Author Éric Vépa evepa@sodius.com |
| | Measure to Table | August 30th , 2007 |

1. ATL Transformation Example: Measure to Table

The Measure to Table example describes a transformation from a Measure model to a Table model.

1.1. Transformation Overview

The aim of this transformation is to represent measurement data as generic tables. This model will be used by other presentation transformations.

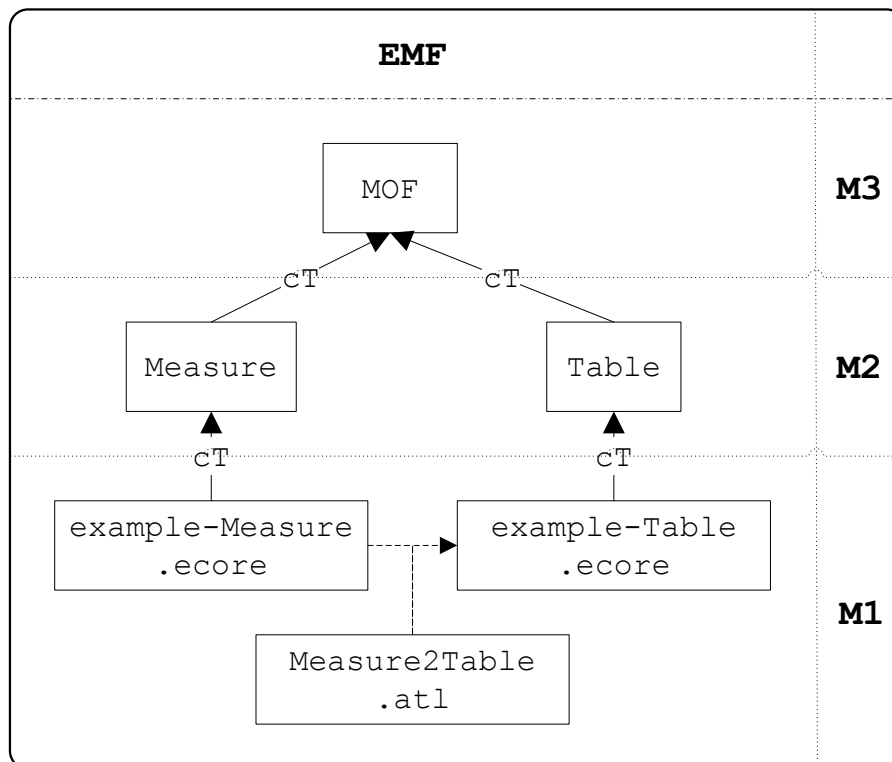


Figure 1: Overview of the transformation

The input model contains measurement data collected by other measurement transformation. For each kind of model element measured a table is created, with all metrics defined for this model element.

2. Metamodels

2.1. Measure

The Measure meta-model is used to stored the data collected after a model measurement.

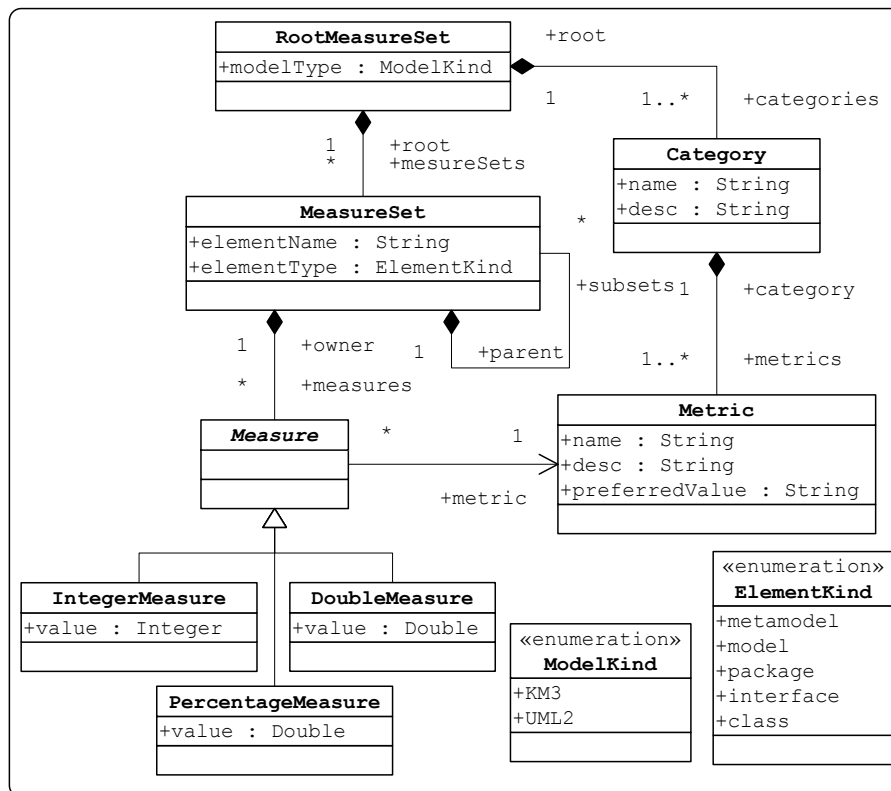




Figure 2: Measure meta-model

A measure model is in the following way made up: the root is a set of measure (*RootMeasureSet*) which contains information on the type of measured model (*modelType* among *KM3* or *UML2*), a set of categories of metric and sets of measure for each model element measured. A category (*Category*) corresponds to a metric set with a *name* and a description (*desc*) (an acronym and its definition). A category gathers one or more metric (*Metric*) also defined with a *name* and a description. A default predicate is also associated (*preferredValue*), it is the desired values for the metric (for example $\neq 0$ or > 75). A set of measure (*MeasureSet*) described measurements performed on a model element (*elementName*) of a given type (*elementType* among *meta-model*, *model*, *package*, *interface* or *class*). The set of measure are structured between them, for example a set of measure on a package will contain the set of measure of the classes that this package contains. A measure (*Measure*) is associated to a metric and is declined in several versions. Measures with an

| | | |
|--|-----------------------------------|---|
|   | ATL Transformation Example | Author Éric Vépa evepa@sodius.com |
| | Measure to Table | August 30th , 2007 |

integer, real or percentage value (respectively *IntegerMesasure*, *DoubleMeasure* and *PercentageMeasure*).

2.2. Table

The source metamodel of Table is described in Figure 3 and can be found in the Atlantic Zoo [2].

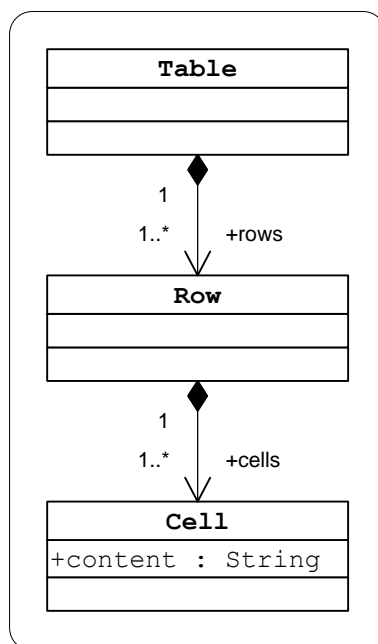


Figure 3: Table Metamodel



Within this metamodel, a Table is associated with a Table element. Such an element is composed of several Rows that, in their turn, are composed of several Cells.

3. Transformation from Measure to Table

3.1. Rules specification

These are the rules to transform a Measure model to a Table model.

- For each not empty sequence of MeasureSet of the same element type, the following element is created:
 - A Table element.
- For each Table element created, the following elements are created:
 - A header Row element.

| | | |
|--|-----------------------------------|---|
|   | ATL Transformation Example | Author Éric Vépa evepa@sodius.com |
| | Measure to Table | August 30th , 2007 |

- Several value Row elements.
- For each header Row element created, the following elements are created:
 - A first Cell element with the kind of the model element measured.
 - Several Cell elements with the name of each metric defined for this kind of model element.
- For each value Row element created, the following elements are created:
 - A first Cell element with the name of the model element measured.
 - Several Cell elements with the value of each metric defined for this kind of model element.

3.2. ATL code

This ATL code for the Measure2Table transformation consists in 2 helpers and 6 rules.

The transformation uses the library defined in section 4.

The attribute helpers *modelKind* and *elementKind* are a mapping between the corresponding enumeration and the desired string value.

The entrypoint rule *Tables()* creates a Table element for each not empty sequence of MeasureSet of the same element type.


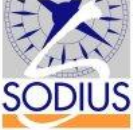
The called rule *AllMeasureSet2Table* allocates a Table for a sequence of MeasureSet of the same element type. The rule creates a Table element (“table”) which is composed of a header Row element and several value Row elements.

The called rule *MeasureSet2HdrRow* allocates a Row for a MeasureSet. The rule creates a Row element (“hdrRow”) which is composed of a first Cell element with the kind of the model element measured and several Cell elements with the name of each metric defined for this kind of model element.

The called rule *MeasureSet2ValueRow* allocates a Row for a MeasureSet. The rule creates a Row element (“valueRow”) which is composed of a first Cell element with the name of the model element measured and several Cell elements with the value of each metric defined for this kind of model element.

The called rule *Sequence2Row* allocates a Row for a sequence of Cell elements. The rule creates a Row element (“row”) which is composed of the sequence given.

The called rule *String2Cell* allocates a Cell for a string. The rule creates a Cell element (“cell”) which content is string given.

| | | |
|--|-----------------------------------|---|
|   | ATL Transformation Example | Author Éric Vépa evepa@sodius.com |
| | Measure to Table | August 30th , 2007 |

4. ATL Library MeasureHelpers

4.1. ATL code

This ATL code for the MeasureHelpers library consists in 18 helpers.

The two helpers *metricName* returns the metric name prefixed by the name of the category of the metric.

The helper *allMetrics* returns all the metrics which category name is given.

The helper *absoluteName* returns the absolute name of the measured model element for this measure set.

The helper *allMeasureSets* returns all the measure sets for a given element type and sorted by element name.

The helper *categories* returns all the categories used by the measures of the measure set.

The helper *allMeasures* returns all the measures of the measure set and sorted by metric full name.

The helper *allMeasures* returns all the measures of the measure set for a category.

The helper *getModelKind* returns the String value for model type of the root measure set.

The helper *simplifiedValue* returns the value with at most one digit after the dot.

The helper *getElementKind* returns the String value for element type of the measure set.

The helpers *stringValue* returns a string value for a measure with an integer, a real or a percentage value.

The helper *toNumber* returns a real value for a percentage measure.

The helper *matchPreferredValue* returns true if the value match the predicate for the preferred value.

5. References

- [1] ATLAS (ATLantic dATA Systems) Official Webpage: <http://www.sciences.univ-nantes.fr/lina/ATLAS/>
- [2] The Atlantic Zoo: <http://www.eclipse.org/gmt/am3/zoos/atlanticZoo/>